Worcester Polytechnic Institute
Digital WPI

Interactive Qualifying Projects (All Years)

Interactive Qualifying Projects

April 2017

COMPUTER SCIENCE EXPERIENCE AT WPI

Christopher Scott Griffin
Worcester Polytechnic Institute

Jimmy Ngoc Tran
Worcester Polytechnic Institute

Steven Charles Yevchak
Worcester Polytechnic Institute

William J. Craft
Worcester Polytechnic Institute

Project Number: CEW-1702

COMPUTER SCIENCE EXPERIENCE AT WPI

An Interactive Qualifying Project Report:

submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by

William Craft **Christopher Griffin**

Jimmy Tran **Steven Yevchak**

Date: March 7, 2017

Approved:

Professor Craig Wills, Advisor

Abstract

This report addresses issues related to confidence in Computer Science, and how it affects performance and the decision to take Computer Science courses. Our aim with this project was to gather data on this issue broadly, and specifically at WPI, to determine how much confidence affects people and what has been done to mitigate this issue elsewhere. A survey was distributed to WPI students to collect data on how confidence affects Computer Science of WPI. Our report concludes with several solutions that worked at other schools to improve performance and diversity. It was found that confidence in computing ability varied by gender, ethnicity, and major where the minority group or groups in each of those categories had lower confidence in their computing ability. Those who have had prior practices in the computing field had a statistically significant increase in confidence versus those who have not. Females also had a significantly lower confidence than their male counterparts and especially when comparing themselves to everyone else.

Contents

1. Introduction

Diversity is lacking in many STEM fields, and Computer Science is no exception. Computer Science is still a young field, and there is still much debate about how it should be taught. Programming courses attract far more males than females.

The studies we examined pointed towards confidence being a big issue in Computer Science classes. Many students perceived their peers to be more confident than them, and this hurt their performance. This perceived confidence gap is more pronounced in minorities, partially because of their underrepresentation in the field[14]. This gap was shown to be even more severe in between the two genders, with females believing they were even further behind their peers[14]. The studies we looked at attributed this largely to males being naturally more competitive, and a general nerd culture which was more appealing to males.

After analyzing studies conducted at other Universities, several common issues were identified, and a survey was created to determine how much these issues affect Computer Science students at WPI. Our survey contained questions about demographic, experience, and confidence. The survey was geared to determine how students' confidence in their own abilities, and their peers abilities affect performance, and also to determine what factors affect a student's confidence. Our survey was designed to include questions about how confidence changed over their time at WPI. We devised hypothesis questions to focus the survey, and attempted to answer them in our analysis. Our survey found that confidence was a factor that affected students at WPI, and the main factors that affected this were experience and perceived confidence of peers. The survey was distributed via email to WPI students of different majors. The results of our survey, along with the information collected from studies about what other schools have done to address these issues, can be used to better shape Computer Science curriculum and teaching style at WPI, and increase performance and diversity.

The report follows is organized as follows. Chapter 2 is our background chapter, we researched issues and studies related to our project. In Chapter 3 covers the design of our survey, and Chapter 4 covers how we implemented that design. Chapter 5 lists the results from the survey we distributed. In Chapter 6 we analyzed the data and drew conclusions about it. Chapter 7 is a summary of our analysis. Chapter 8 discusses future work related to this project. Finally Chapter 9 is our conclusion.

2. Background

We examined articles that are related to performance and diversity in Computer Science. Researchers at many universities performed these studies, and we quickly identified that a major common issue that had been identified was confidence. Most of the studies found mentioned confidence in some form or another, and many studies focused on it.

2.1 Studies related to common issues and introductory courses in computing.

Our preliminary research turned up many commonly identified issues with how Computer Science is being taught at other universities, and what steps have been taken to fix them. Recently many universities have reported high dropout rates and problems grasping the basic programming concepts in the first programming courses[1]. Students are assigned work and reading from the module at a level appropriate to the objectives of the long-term goals of their major. It has been estimated that more than two million students started computing studies in 1999 and 650,000 of them either dropped or failed their first programming course. Some universities were able to lower these dropout rates and raise average grades through revised course infrastructure. The analyses of these data indicated three main problems in the course: programming discipline difficulty, course arrangement complexity, and limited student motivation.

Introductory courses in programming to non-engineering students should focus on addressing a few key points[2]. Mitigating the problems that arise from a desire to fit everything into one quarter, causing some material to be rushed or skipped. Using the opportunity to encourage students to explore programming further, and give students useful tools for use in their respective fields. A novel introductory computer science course for non-majors was implemented. Students focus on algorithms, the principle of computational thinking, use a flowchart simulator to experiment with various short algorithms, and build simple computer games without dealing with programming language syntax. These classes have had steadily increasing enrollments and interest from various departments on campus which indicates that this course has become a successful addition to introductory Computer Science offerings.

2.2 Studies related to gender and ethnicity in computing.

There was a study performed on the experience of female computer science undergraduates at CMU with a focus on why they choose to stay, or stray from the field[3]. A lot of reasons stem from social norms, affecting both the way people treat them, and the way they feel about themselves. Many females feel self-doubt when comparing themselves to their male counterparts despite excelling in the field themselves, and many grow up with limited exposure to CS, and as a result feel like they are behind from the start.

Minority students are more likely to be intimidated by classes that are rumored to be difficult[4]. At many schools faculty do not take appropriate steps to make Computer Science as race inclusive as it can be. Even slightly different treatment by a professor towards a minority student can cause the student to feel disparaged and not expected to succeed. Teaching Assistants need training in communicating in English. While they work hard, foreign TAs can have difficulty properly communicating with their students.

A study showed that eliminating the 'nerd' culture from Computer Science benefits women, minorities, and even the 'nerds' themselves[5]. This study concludes that "blatantly countering 'nerd' stereotypes may cause more harm than help". Studies have shown that women have the same enthusiasm for computers as their male peers, but lack confidence and comfort. When asked how comfortable they felt, Women rated themselves similarly to their male counterparts, but when asked how they viewed their peers, they rated them as more comfortable than they are. Women rated themselves an average of a half point (scale of 1-10) less confident than their peers, while men rated themselves 6/10s of a point more confident. All students in the cohort described time, or coding speed, as the fundamental measure of ability. The study suggested that in our gender system, men have an incentive to spin their struggle into a rite of passage and endurance. Women in our culture do not experience the same expectation, and the Computer Science culture is more geared towards the male approach. “I just want to make sure I’m making the right decision. I like CS. I can’t see myself doing anything else. I’m not interested in other things. But in terms of ability, I’m not sure I’m there.”

It is clear that there is a fairly large gap in the number of males and females in Computer Science courses. A number of variables have been proposed to account for this gender difference. The first is likely an issue in all STEM subjects, traditional socialization practices reinforce math and science as mainly male domains. On this studies have found lower confidence ratings and greater math anxiety among women[6]. There is a belief that computer science is a more competitive, alienating field that further discourages women, who are not generally as competitive as their male counterparts, from pursuing careers CS. Ways of mitigating and combating these variables have been proposed and tested. Pair programming, when used as a form of collaborative learning, has been shown to increase the number of women (and men) persisting in their previously stated intent to pursue degrees in computer science. It has been found that women who program in pairs will have higher retention rates than women who program independently. A comparison of students who used pair programming with those who did not indicated that pairs were significantly more likely to remain in the course through the final exam (90.8%) than were non-pairs (80.4%)[6].

2.3 Studies related to non-computing majors taking computing courses and tailored courses.

At Georgia Institute of Technology, Atlanta, where introductory Computer Science courses are a requirement for Computer Science majors and nonmajors alike, two tailored introductory courses were introduced as an alternative to the traditional course[8]. The results were encouraging: more nonmajors succeeded (completed and passed) in tailored courses than in the traditional course, students expressed fewer negative reactions to the course content, and many reported that they would be interested in taking another tailored Computer Science course. "Flipped" instruction offers a vision of class meetings devoted to active learning, in exchange for students spending time outside of class acquiring basic knowledge from readings or video lectures.

Peer Instruction (PI) is a teaching method that supports student-centric classrooms, where students construct their own understanding through a structured approach featuring questions with peer discussions. For the same instructor teaching the same course, the study found PI decreases the fail rate, on average, by 67% (from 23% to 8%)[8]. Courses designed specifically for non-majors include motivation, the welcoming environment of computing naïve students, and openness to stereotype myth busting including gender, career, and engagement issues.

2.4 Other Studies

A study conducted at Georgia Tech showed that lots of students are better motivated through tailored courses than traditional general courses in CS[8]. There courses boasted fewer students receiving DWF (D-grade, withdrawal, failure) than those in regular Computer Science courses. Even in courses tailored for engineers there is noticeable improvement, there was a DWF rate of 42.9% in the traditional course, and the tailored course was down to 18.7%. Students were surveyed after taking these courses, and 12.1% of the students responded that they had a negative outlook on Computer Science on CS, and none of the students in the tailored course responded this way.

2.5 Summary

The most common theme we found throughout the studies we read was confidence. Confidence is a major factor in common issues in Computer Science. Students who believed they were less confident than their peers ended up performing worse as a result. Universities that took steps to tailor courses to increase students confidence through methods like pair programming found reduced fail rates, as well as a higher number of students taking additional Computer Science courses in the future.

3. Design of Survey

Non-majors are a growing segment of students who take computer science classes at WPI. Using what is known from the research background chapter, a study on how the non-majors and majors at WPI view the computer science curriculum will be created. It will include questions that explore their experience in the field, their mindset, and their general views prior to arriving at the institution. Their experience in courses specifically for non-majors vs standard computer science courses. Students will also be drawn from the group that chose to enroll in courses as their free electives rather than being mandatory to graduate.

According to our research, what is understood is that a big obstacle for many people is the confidence level that they portray towards the curriculum. Topics of interest for this survey include: the knowledge or level of skill in programming students entered WPI with; their experiences taking the programming courses; their level of confidence before, during, and after taking the courses; and their perceptions of classmates. These questions will help us understand in a more accurate view the current experience in computer science at our institution, and to see differences stratified according to gender, ethnicity, and major. The results will allow the ability to analyze the data and determine if those groups have differing confidence levels, and why. Specifically confidence will be examined by previous Computer Science experience, gender, ethnicity, and the confidence of non-computing majors taking computing classes.

Confidence was chosen to be examined as numerous research papers scoured has confidence as a factor in why students decide to stay in their majors and courses. By examining their confidence through the years, it is possible to potentially pinpoint whether or not it is a factor in the amount of courses they decide to stay enrolled. Furthermore, it could also give an indication of whether or not the number of classes can affect their confidence. This will be important in figuring a solution to help students stay in the curriculum.

3.1 Research questions and the survey questions to match

1. What effect, if any, does a student's confidence in their computing abilities have on their performance in computing classes?
    a. What is your level of confidence in computer ability prior to college courses in computing?
    b. What is your level of confidence in computing ability after taking some college courses in computing?
    c. Do you feel ready to work in the computing field? (another way to measure confidence)
    d. How many years have you been taking college computing courses?
    e. What is your current GPA in computing courses?

2. What factors indicate a student's level of confidence in computing ability before they enter college? Both factors that are inherent to the person (ethnicity, gender) and factors that are the students' previous experience in computing.
    a. What is your gender?
    b. What is your ethnicity?
    c. List experiences with computing prior to college
    d. What is your level of confidence in computer ability prior to college courses in computing?
    e. What is your level of confidence in computing ability after taking some college courses in computing?

3. Do the student's peers significantly impact their confidence in computing, and if so, is there a correlation between this and their confidence/ability?
    a. Compared to my peers my computing ability is?
    b. For computing projects how do you prefer to work?

4. How is a student's confidence in their computing ability affected during college classes? What factors in the classroom positively or negatively affect a student's confidence?
    a. What is your level of confidence in computer ability prior to college courses in computing?

b. What is your level of confidence in computing ability after taking some college courses in computing?
c. What is your preferred style of teaching?

3.2 Summary

One of our goals was to use the data collected to identify what is successful in getting non-majors to take more computer science courses. Similar to the previous goal was to find what in computing course is successful in among minority students (in computing this would be female, Hispanic/Latino, Asian, and African American students. Both of these goals can be done by analyzing these students' preferences in computing classes and recommending those changes to better accommodate them. Two metrics of instance are student's preference in teaching style and in group size when assigned group work. Teaching styles vary by course topic and professor but generally fall into five major categories: lecture based; project based; significant use of examples like live coding; labs; and learning primarily from peers. Finding the preferred teaching style of minority groups and non-majors would be useful in forming recommendations that improve their computing class experience.

Four general research questions were designed and assigned each survey question to one of these research questions. Our research questions were and corresponding survey questions were: what effect, if any, does a student's confidence in their computing abilities have on their performance in computing classes; what factors indicate a student's level of confidence in computing ability before they enter college; do the student's peers significantly impact their confidence in computing, and if so, is there a correlation between this and their confidence/ability; and how is a student's confidence in their computing ability affected during college classes.

4. Implementation of Survey

A survey was developed to encompass as many students as possible at our institution. The questions inquired about their self-projected confidence in their ability in the computing field. It also included questions that hinted at their background, such as history of courses, planned future courses, and even their projected confidence in their ability after finishing their curriculum. The questions in the survey were designed to get a sense of people's experience and background, and evaluate their confidence taking into account these factors.

A lot of the questions at first seemed to target a minority of people, opposite of what was wanted. What was wanted is an evaluation of the curriculum to see why people are shying away or staying, and it would not help to exclude those deterred from it. At first, the term “computer science” was used and too many people, if that was not their major, they would not give it a second glance. Thus, it was changed to “computing ability” to be more inclusive.

At first, questions were simply generated that would gather interesting information relating to these issues, but the survey was fairly disorganized and unfocused. Survey questions were then compared to the research questions listed in Section 4 Design to determine if they were pertinent questions. Those same research questions were also used as a guide to create new questions to fill missing gaps in survey questions that had already been determined to be used. Anywhere from two to five survey questions were decided on to answer the each research questions.

5. Results

This chapter lists all of the questions we included on the survey, and the responses we received. The questions have been thematically grouped based on what information they were designed to collect. The background questions gave us an idea of who was responding to our survey, and allows us in later chapters to determine how background affects responses to other questions. Education questions collect information on how far into their computing studies responders are, and how they are performing. Questions on preferences gather data about confidence and comfort in computing courses.

5.1 Background Questions

These questions establish a background on the respondents. The majority of our respondents are White, Male, and Computer Science and Engineering majors. Our respondents had a mixed exposure to computing before college.

Choose the group that best fits your major

Non-computing Engineering	36	14.4%
Computer Science and Engineering	165	66.0%
Business	1	0.40%
Humanities	0	0.00%
Game Development	3	1.20%
Life Sciences (Chem, Bio, etc)	3	1.20%
Management	3	1.20%
Mathematical	4	1.60%
Physics	0	0.00%
Social Sciences	0	0.00%
Robotics	31	12.4%
Other	4	1.60%

What is your gender?

Male	190	75.7%

Female	53	21.1%
Other	5	1.90%
Prefer not to answer	3	1.20%

What is your ethnicity?

Hispanic/Latino	16	6.40%
American Indian/Alaskan Native	2	0.80%
Asian	40	16.0%
Black/African American	0	0.00%
Native Hawaiian / Pacific Islander	0	0.00%
White	180	72.2%
Unknown	1	0.40%
Prefer not to answer	10	4.00%

Check all experiences with computing prior to college.

Taken a non-AP course in high school.	98	39.0%
Taken an AP course in high school.	94	37.4%
Designed a webpage.	81	32.2%
Self-taught a programming or scripting language.	123	49.0%
Other	36	14.3%
None	51	20.3%

5.2 Education Questions

These questions determine current education status as well as education goals of our respondents. Most of our respondents have taken at least 1-3 computing classes.

Number of computing classes taken so far.(Including classes in progress)

0	10 4.00%
1-3	92 36.8%
3-7	70 28.0%
8+	78 31.2%

Number of computing classes planned to take by graduation.

0	6 2.30%
1-3	33 13.1%
4-7	61 24.3%
8-12	30 11.9%
12+	121 48.2%

What is your current GPA in computing courses?

< 2.0	6 2.40%
2.0 - 2.49	6 2.40%
2.5 - 2.99	20 8.10%
3.0 - 3.49	73 29.5%
3.5 - 4.0	125 50.6%
> 4.0	17 6.80%

5.3 Questions on Preferences

These questions helped us to determine trends in confidence and preferences in different sections of the population. The confidence changes between our respondents were examined as well as how our respondents prefer their classes to be structured.

What are some things about your peers that affect your confidence in your computing ability?

Their confidence	96 38.2%
Their knowledge	183 72.9%

Their complexity of coding	133 52.9%
The way they portray themselves	82 32.6%
Other	19 7.50%
I don't get affected by my peers	45 17.9%

For computing projects how do you prefer to work?

By myself	108 43.0%
With a partner	108 43.0%
A group of 3 to 5 people	34 13.5%
Groups of 6 or more	1 0.40%

What is your preferred style of learning?

Lecture based	25 9.90%
Mostly examples(live coding)	78 31.0%
Project based	97 38.6%
Hands-on (Labs)	41 16.3%
Through Peers	4 1.50%
Other	6 2.30%

On average how many computing courses were taught by your preferred style of teaching?

Less than 25%	62 24.7%
25% to 50%	81 32.2%
50% to 75%	68 27.0%
75% to 100%	40 15.9%

5.2 Summary

The results from this chapter was analyzed in the next chapter to draw conclusions based on the data. The survey received over 200 responses and collected a lot of data. We used the data to answer our hypothesis questions.

6. Analysis

When taking a closer look at the data, it shows the mean of level of confidence among all of those who have taken this survey broken up into different sub-categories, such as by major, gender, and other qualifications, prior to joining college. Survey takers have ranked their own confidence levels prior to college, their current level, and where they expect to be after graduating, on a scale of 1 to 10, where 1 is least confident and 10 being most confident in their computing abilities. 95% confidence intervals are computed to determine if the difference between particular groups is statistically relevant or not. Low numbers of participants from some groups selected led to large confidence intervals especially when combined with the high variance that was seen particularly in the prior to college confidence in computing ability. Most of the results are largely unsurprising as the background studies led us these hypotheses.

6.1 Participant Confidence Overall

Confidence increasing as time goes by is a trend that is seen continued over all the data collected for each group there typically is a statistical increase in students' confidence from prior to current to expected. However this trend is not always the case even though the average confidence of a group is always increasing the group size may be small enough with a large enough variance that it is not possible to say with a 95% confidence interval that there is difference between the same groups at prior, current, and expected.

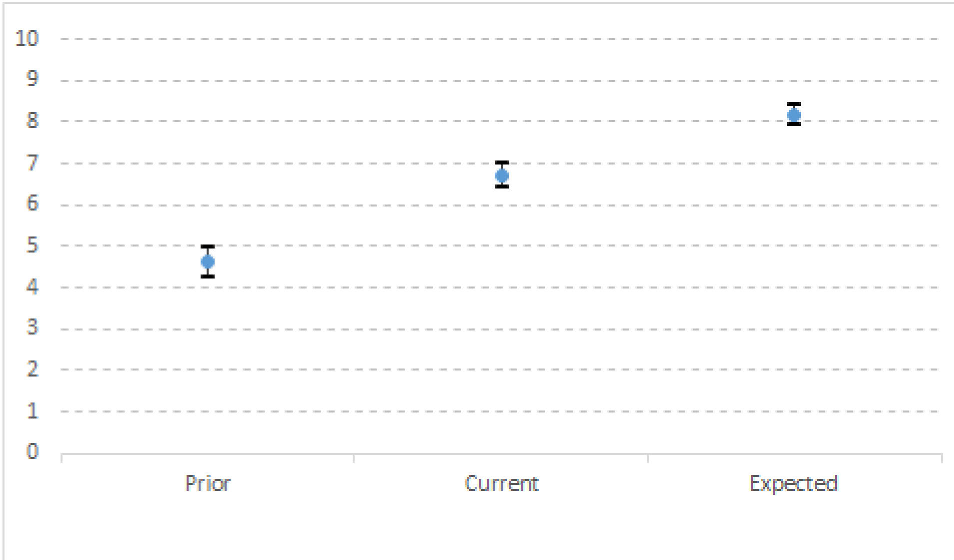


Figure 1: 95% confidence intervals of all participants at their prior, current, and expected levels of confidence.

6.2 Confidence by Majors

At a quick glance of the data, it is not surprising to see that those in the computing majors have a higher mean for starting level of confidence. It is expected that confidence is computing ability is rising over time and that it indeed happening as shown in Figure 2. What also is not too surprising is that the people who are in the fields that require computing are generally more confident than those that are outside of the computing field prior to starting off in college. Note that these three majors are chosen based on having a high number of responses while others are omitted due to having a low number of responses. With a 1.7 score difference in the average confidence level, it is evident that those who are in the major feel more secure in their ability. This gap widens even further when asked their current level of confidence, rising to roughly a 1.9 in a score difference. Once again, when asked to gauge what they think their computing ability are projected to be when they graduate college, the gap widens a significant amount. A whopping 2.4 score difference between the two, with computing engineers having a much higher projected confidence at 8.6 compared to 6.2 of the non-computing engineers. The trend between non-computing and robotics majors do not change drastically when going through the three phases of where confidence is measured. However, computing shows a significant change in confidence levels in all three measurements of time. There is statistical difference in the levels of confidence when comparing the past, the current, and the expected. The gaps between the upper limit and the lower limit of the following time-frame is not close to each other.

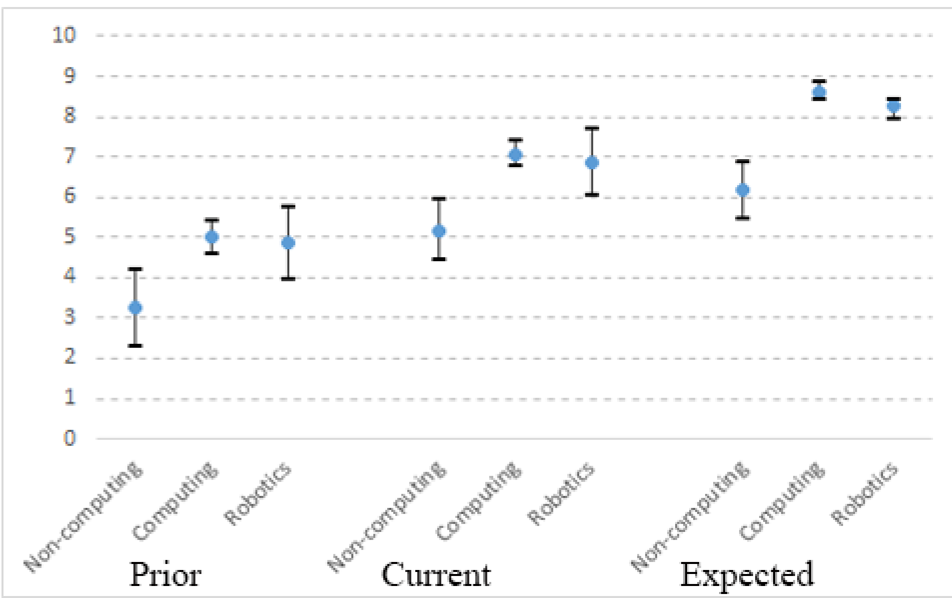


Figure 2: 95% confidence intervals of majors at their prior, current, and expected levels of confidence.

Those in the robotics major start off near the same as the computing majors. With a difference of about 0.1 with robotics being the lower confident major according to the results, they are pretty average in terms of their belief in their abilities. When asked about their current level of confidence, they too also match up well with the computing major. With only a difference of 0.2, again robotics being the lower, they are both fairly confident (6.9 for robotics and 7.1 for computing) in their abilities in computing. However, the gap does widen a bit when asked about their projected confidence. The gap rises to a 0.4 score difference, again with robotics being less confident.

6.3 Computing Activities

Computing related activities done in high school also had unsurprising effects on confidence levels of students beginning college as shown in Figure 3. Students who did no computing related activities were almost 2 to 3 points of confidence behind students who did at least one computing activity during high school. 20.5% of people surveyed reported having done no computing related activities during high school and entered college with an average confidence of 2.1 the lowest of any group that was measured. While students who during high school gained computing related experience either from classes AP or otherwise where within 5 to 6 points of confidence with participants who took non-AP classes falling about half a point lower. Those students doing computing in their free time during high school whether they were learning computing languages or building webpages also entered with a confidence of 5 to 6 points. This confidence gap between these remains decreased a bit with current confidence levels where students who entered with no experience had 5.8 points confidence and those with experience ranged from 7.0 to 7.5 points of confidence. The gap was nearly halved after students took college computing courses. Expected confidence close the gap a bit more with students who had no prior experience averaging 7.5 points of confidence and students with previous experience ranged from 8.3 to 8.7. The activities that in order that led to the highest levels of confidence were self-taught a language, designed a webpage, taken an AP course, then taken a non-AP course. It shows that doing one of these activities generally boost confidence overall.

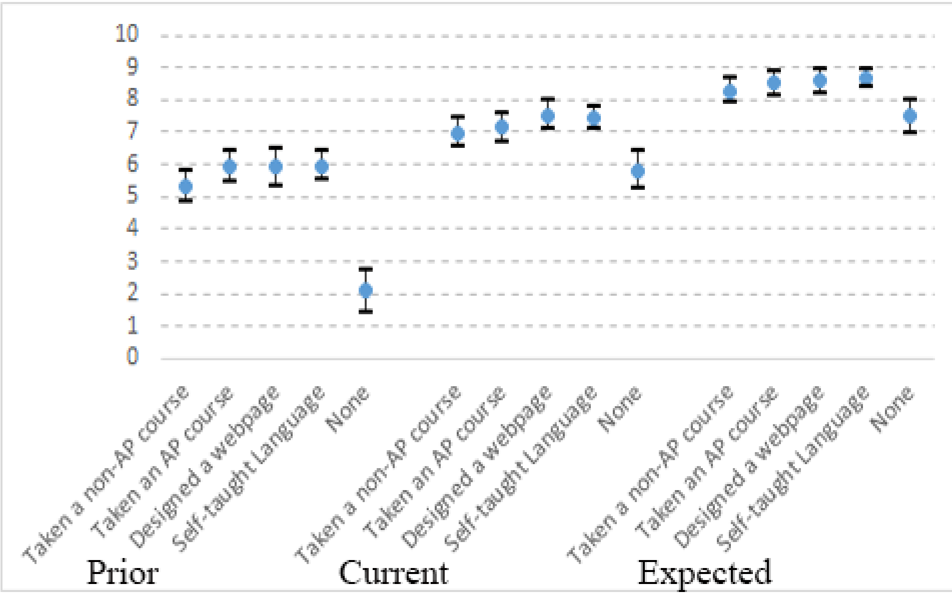


Figure 3: 95% confidence intervals of how pre-college activities affect their prior, current, and expected levels of confidence.

6.4 Confidence by Gender

The third lowest category of people who have the low scores for overall confidence shown in Figure 4 is females who started with an average confidence of 3.3 while males started with a confidence of approximately 5 which is among the higher starting confidences. This is not surprising as the background included prior have shown studies to believe confidence levels between would have a disparity like the one present. This one and a half point confidence gap grows to current levels of confidence where females average 5.3 points of confidence and males average 7.1 points of confidence. Expected levels of confidence draw the two groups closer together with females averaging 7.2 points of confidence and males averaging 8.4 points of confidence. The trend shown here is that males tend to have a statistically higher confidence level than females do overall. The intervals between the two genders do not intersect at all when being compared.

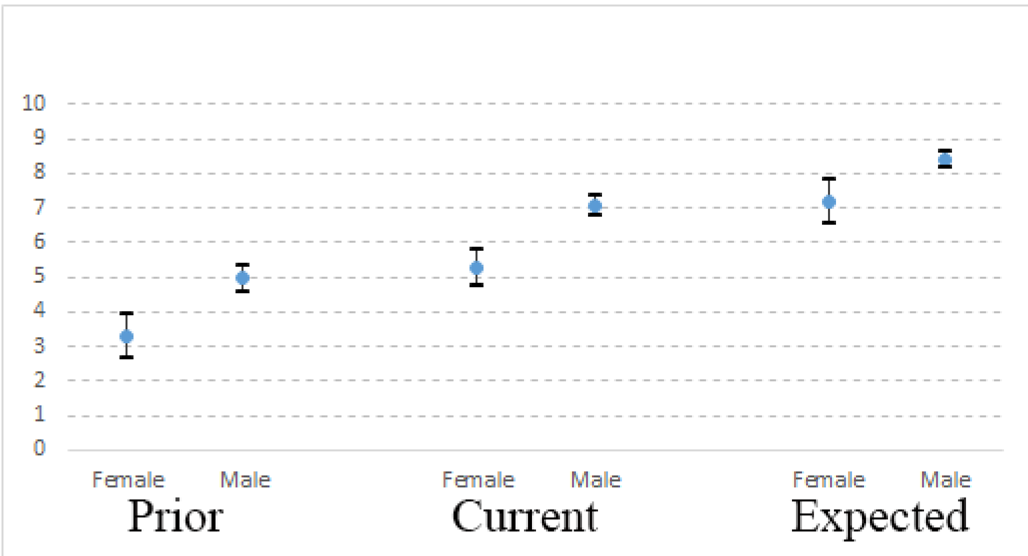


Figure 4: 95% confidence intervals of females and males at their prior, current, and expected levels of confidence.

6.5 Confidence by Ethnicity

Different ethnicities also had different levels of confidence with Whites being the most confident in their computing ability on all three measures as shown in Figure 5. Again, note that these three ethnicities were chosen due to having high amount of responses. Whites started with an average confidence of 5.0 while Asians began with 3.4 points of confidence and Hispanics began with 3.8 points of confidence. This gap between the three groups stays to current levels of confidence 7.0, Asians had 5.3 points of confidence, and Hispanics had 5.8 points of confidence. Expected confidence it quite interesting as Asians report having higher confidence than Hispanics and closes the gap on the level of confidence that White people reported. White with 8.3 points of confidence, Asians averaged 7.8 points of confidence, and Hispanics had 7.3 points of confidence.

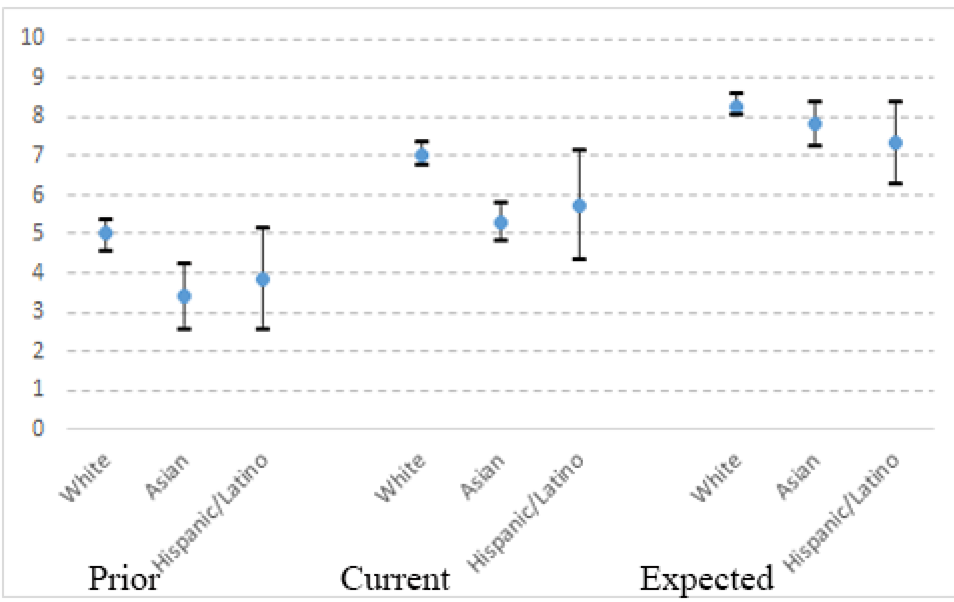


Figure 5: 95% confidence intervals of different ethnicities at their prior, current, and expected levels of confidence.

6.6 Confidence by Number of Courses

Those who have taken numerous courses currently started off with a much higher confidence level than those who have not taken too many yet as shown in Figure

6. Those who took no courses have a much lower average level of confidence at 2.1 on a 1 to 10 scale. Going up the list, those who have taken 1-3 courses have a mean of 4.2, 3-7 courses having 4.6 and those who have taken at least 8 courses with a mean of 5.1.

When checking these groups currently, what is actually interesting is the fact that those who have taken no courses are somehow more confident than the group that has taken 1-3 courses. With a difference of 0.3 in score with those taken 1-3 courses being lower, those who no courses have more confidence. Not surprisingly, those who have taken 3-7 courses have a much higher confidence level at 6.8, a tremendous 1.0 difference in confidence than those who have taken no courses. Last on the list is those who have taken 8 or more courses having a 7.9 score in confidence, 1.1 more than those who have taken 3-7 courses, and 2.1 more than those who have taken no courses. A trend appears that shows those who have taken only 1-3 courses, jump only a small amount of confidence when compared to those who have taken more. Those who have taken more than 3 courses, seem to have a higher confidence jump especially from when they first started taking the courses. Strangely, however, those who have taken more than 8 courses do not really expect themselves to be more confident in the future.

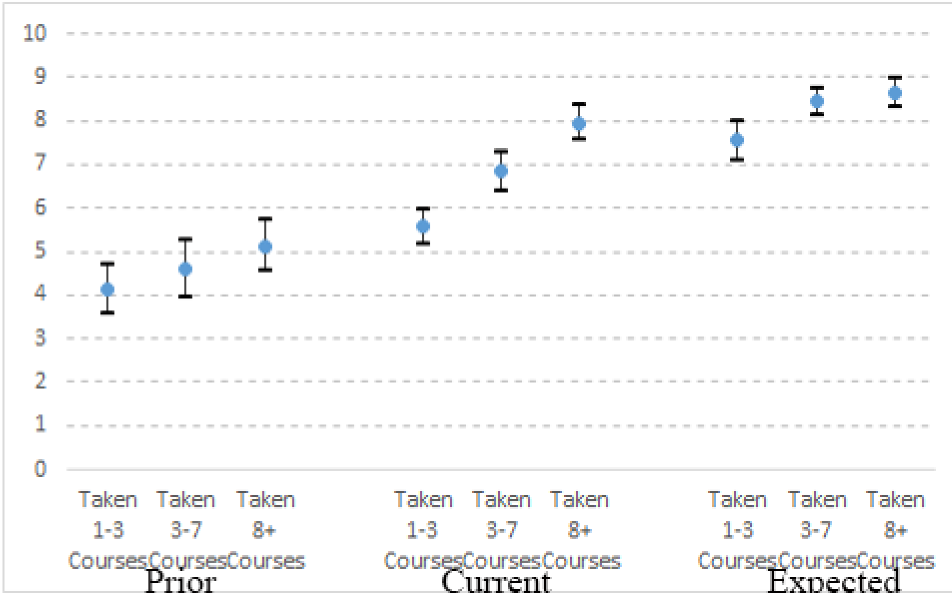


Figure 6: 95% confidence intervals of different number of courses taken at their prior, current, and expected levels of confidence.

6.7 Perceived Confidence

The survey asked for the student's perceived computing ability which followed closely the trends seen in current confidence. However there are only a few groups that with 95% confidence show statistical difference as shown in Figure 7. Between majors there is no statistical difference even though both computing engineer Majors and robotics majors are both a full point higher than non-computing majors. Gender did show a statistical gap with the averages being more than two points apart and the difference between the top of the female confidence interval and the bottom of the male interval being more the a point. For ethnicity statistically white perceived computing ability is higher than Asian. However due to a small sample size of Hispanic and Latinos there is no statistical difference between them and the other ethnicities. For activities done before high school all of them showed a statistical increase over the group of student who took no computing related activities prior to entering college. For students looking to go in computing majors starting some computing activities be college is highly recommended. For courses taken one's perceived computing ability compared to their peers rises there is a statistical increase between students who had taken one to three courses and those had taken three or more. However there was not a statistical increase between those who had taken eight or more courses and those who had taken three to seven.

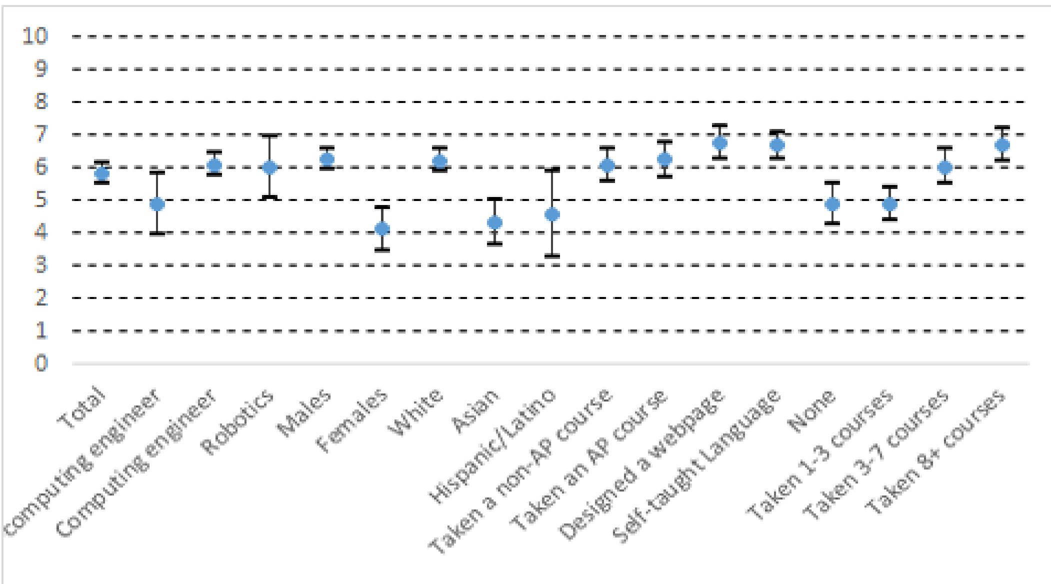


Figure 7: Current computing ability compared to peers.

6.8 Preferred Learning Type

Students have different preference on how they want courses to be taught. These fall into five main categories: lecture, examples(live coding), project based, labs, through peers. Figure 8 shows the breakdown of these choices by gender. The two most preferred types of course structure are those that are project based at 38% of students and those that include mainly examples which in computing is typically done by live coding which had 30% of students. Both were top two choices of both genders however project based courses were preferred by males and example based courses were preferred by females. Most computing courses are obviously going use a mix of teaching styles and some lecture is almost always going to be necessary. However assigning projects which typically have extended time to work on and higher weight in grading are preferred along with class time spent showing how to do what was taught. Live coding also typically includes giving students access to code used in class which helps with further understanding as a study material. Both males and females wanted labs at about equal percentages at around 15% of students. The two least like styles of teaching are lecture and peer learning form PLAs and SAs with only around 10% of students preferring that method of learning.

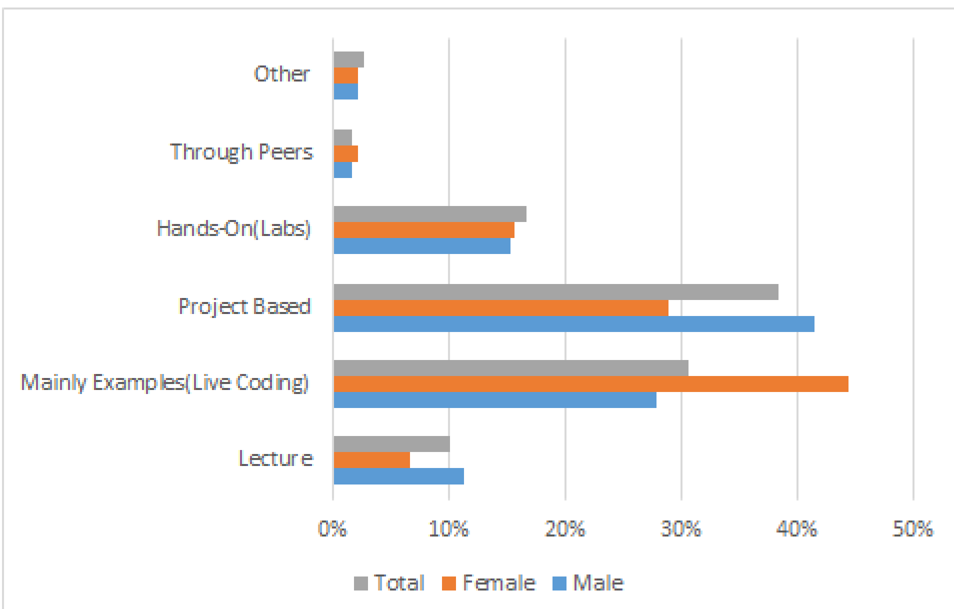
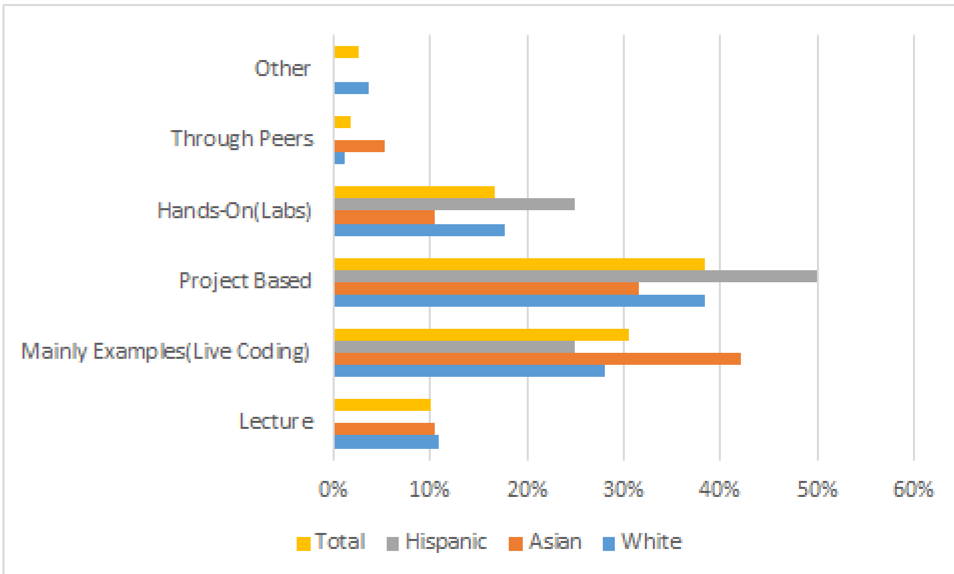


Figure 8: Preferred course style by gender

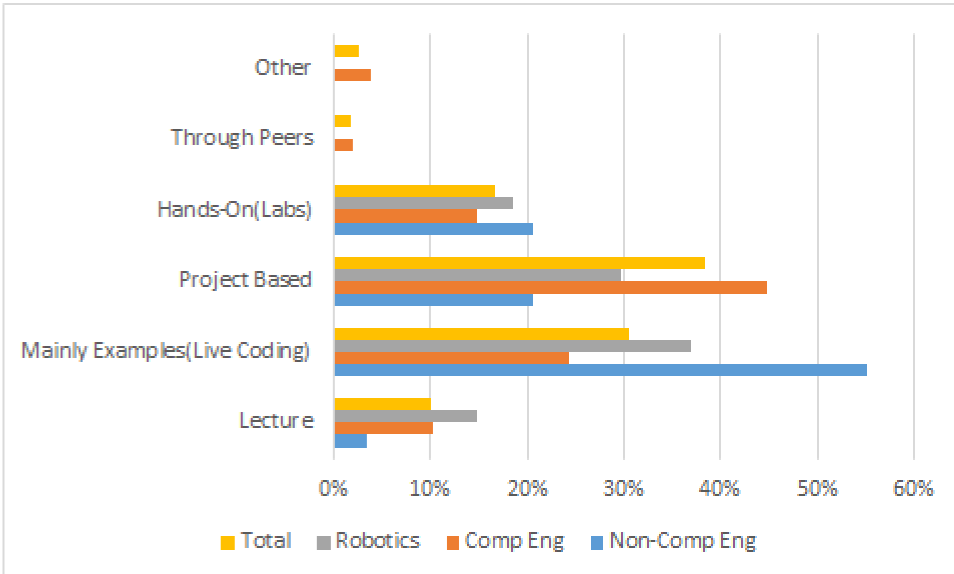
Analyzing students preferred course teaching method by ethnicity only again confirms the two most popular teaching methods are project based and example based. There were only a few abnormalities that Figure 9 shows us the first being that that is that the Hispanic group had a tie for second most preferred teaching method between labs and examples based. Every group also choose project based as their preferred course teaching style with the exception of Asians. The rest of the results were rather



unsurprising and fell in line with the totals of students surveyed.

Figure 9: Preferred course style by ethnicity

Preferred course teaching style by major also has mostly unsurprising results as shown in Figure 10. The same two teaching style were preferred in project based and example based. Both robotics and non-computer engineers preferred example based while computing majors preferred project based. Interestingly computing engineers were enough to push project based to be the overall preferred teaching style. So it would seem the having computing courses that used more examples like live coding during class time would benefit non-majors. Non computing majors also showed the strongest dislike of lecture based courses of any on the groups discussed. The rest of the results show majors matching each other just about evenly for the other course



teaching styles.

Figure 10: Preferred course style by major

6.9 Preferred Style of Working

In addition to how these students prefer to be taught, they have a preference in how they want to learn. Students had the option to choose one of three options: By themselves, with a partner, or with a group of 3-5 people. Figure 11 illustrates their responses into sections based on their choice and their gender. As evident in the graph, the preferred style of working is either with a partner or by their lonesome. While females dominates in wanting to work with just a partner, males are predominantly in between working with a partner or by their lonesome. A lot of intro level courses are generally allotting the students a choice of either just a partner or by themselves, while only some higher level courses involve students working in numerous groups.

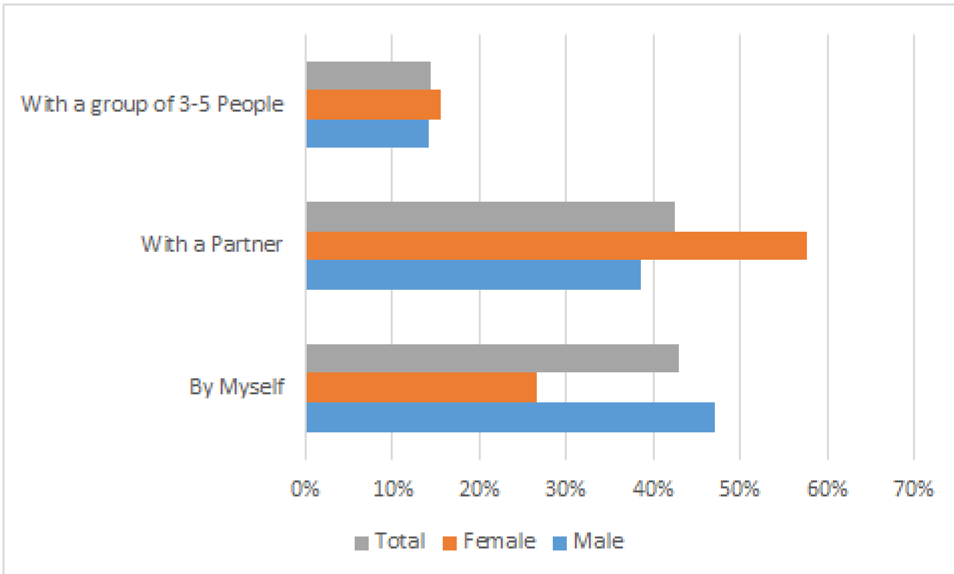


Figure 11: Preferred style of working by gender

When categorized by their ethnicities, the two predominant choices of working is with a partner or by their lonesome. As can be seen in Figure 12, Hispanic and Asian students have chosen working with a partner as their preferred style over the other two. White students seem to be split among with a partner and by themselves with only a small fraction choosing to work in a group of people, alongside Hispanic and Asian students as well. It is not as surprising as working in small groups of people do generally allow for less conflict between everyone involved in terms of how to do assignments.

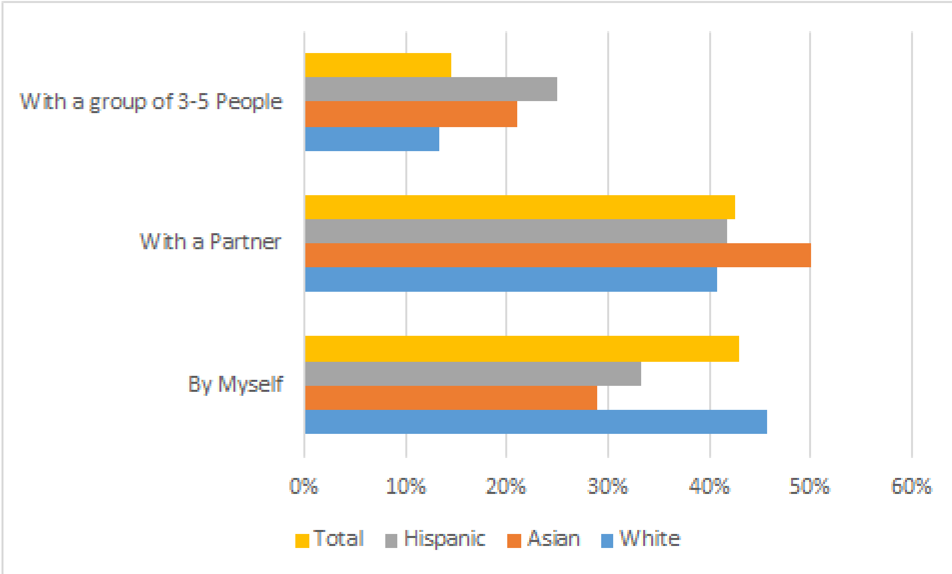


Figure 12: Preferred style of working by ethnicity

Analyzing the results by the majors also yields unsurprising results. This time, the total amount of people are split almost even for working with a partner and by themselves. All three majors have chosen those two options as their preference for learning in their schools. Again, it seems that working with just a partner or by themselves will allow them to work more comfortably than working with a larger group of people. This is surprising still for computing engineers and robotics as the field generally means working with a large team in order to complete tasks.

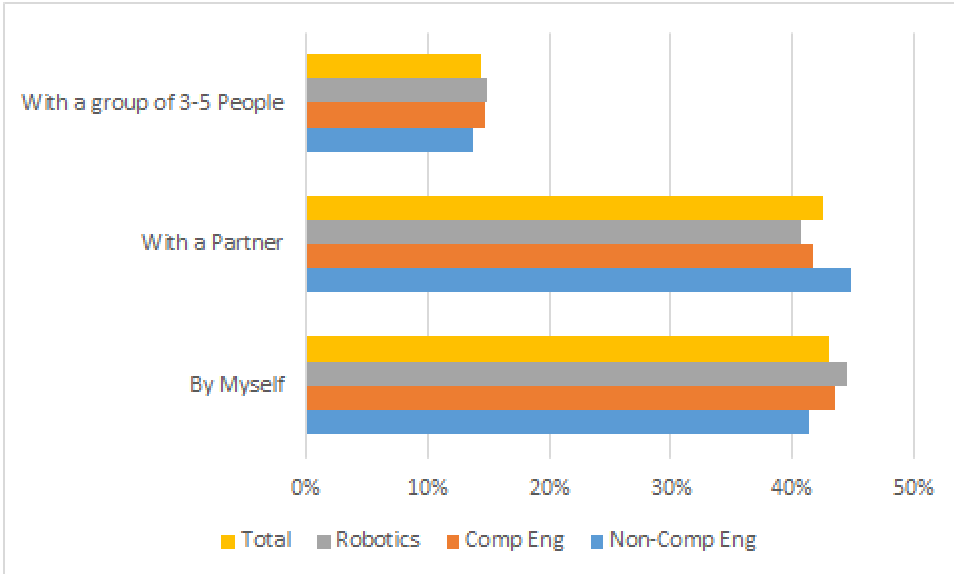


Figure 13: Preferred style of working by major

6.10 Courses Taught in Preferred Style

The number of courses that students are taught by their preferred method of teaching is shown in Figure 11 by percentages of total computing courses they have taken. It appears that a large percentage of students around 46% are regularly taught computing courses by the teaching style they prefer. Regularly meaning that more than half their courses which appears to be quite good considering that there were four preferred teaching styles: Project based; examples (live coding); labs; lecture. However since most courses normally incorporate at least two of these teaching methods that number seems a bit low. Also, unfortunately 54% of students indicate that less than half of their courses are taught by their preferred teaching method. Figure 8 indicates that is a course includes regular project and regular use of live coding demonstrations or other examples relevant to the course that at least 60% of students would satisfied by the teaching methods.

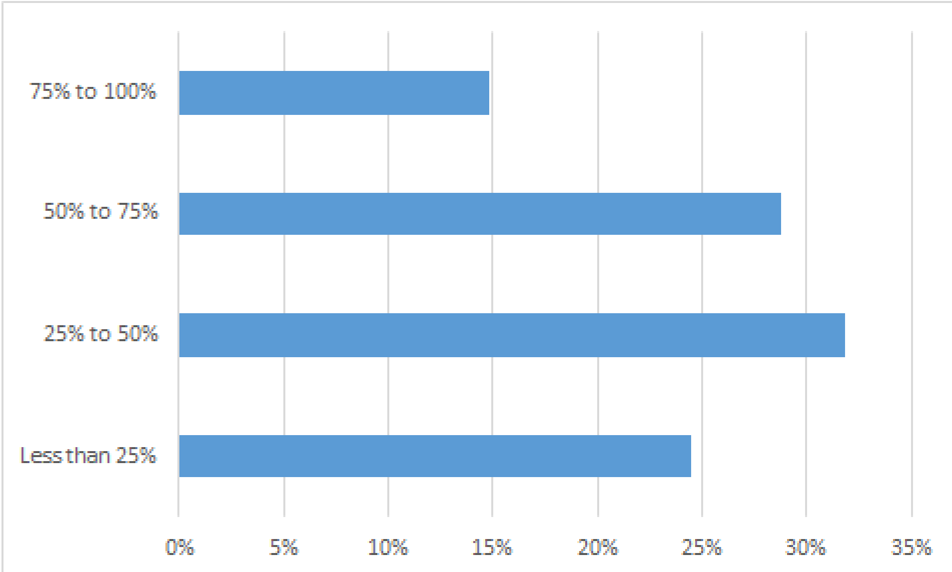


Figure 14: Number of students who indicated that the above number of courses were taught by their preferred teaching style.

6.11 Effect of Peers on Own Confidence

There is a lot of reasons why a person’s confidence can be affected by their peers. As shown in Figure 15, one of the biggest impacts to a person’s confidence is the knowledge that their peers radiate compared to themselves. Approximately 32% of people say that their peers’ knowledge of the subject affects their own confidence of their own computing ability. The second largest factor that affects a person’s confidence is how their peers code their programs. To the person, their peers seemingly create more complex programs than what they are used to viewing. Finally, the way that peers portray themselves, such as their confidence in their code and voice, will make a person feel less confident in their own ability most likely when comparing themselves, especially when the person in question is already not too confident.

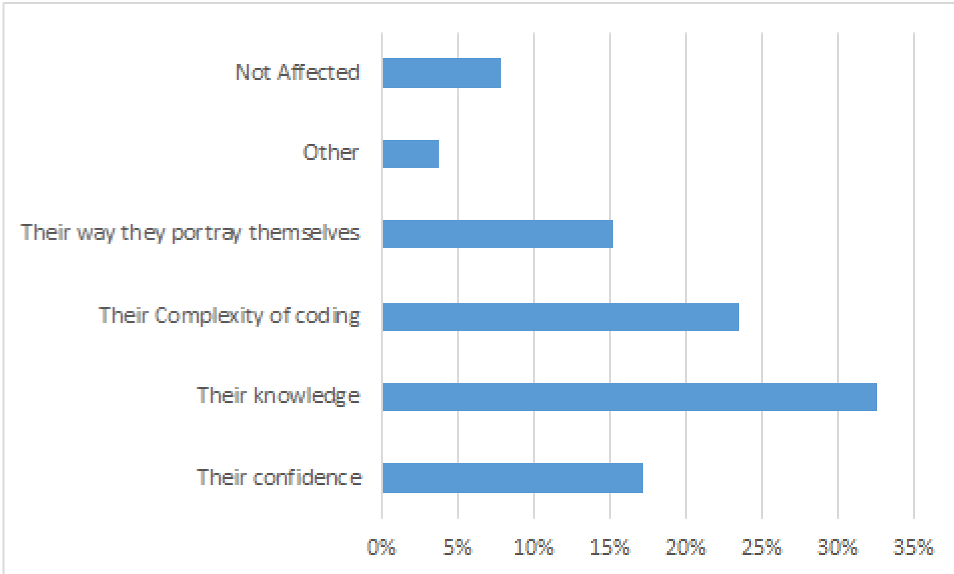


Figure 15: Number of students who indicated that their confidence was affected by their peers in each of the above ways.

7. Summary of Research Questions

The data analyzed showed that there are differences in students' confidence in their computing abilities on the basis of major, gender, ethnicity, prior experience, and number of courses taken. It shows that generally with 95% confidence each of these groups confidence increase from prior to current to expected levels of confidence. The groups with the highest overall confidence in their computing ability were computing majors, White students, and those who had done some computing related study or activity prior to taking college computing courses. Interestingly students who had taken more computing courses showed higher prior levels of confidence possibly indicating skewed memory of true prior confidence for students who had taken 3 or more computing courses. The groups of students with the lowest levels of confidence include non-computing majors, females, Hispanic/Latino students, and those who had done no computing related study or activity prior to college courses. Perceived computing ability followed the same trend as those described above.

Students were asked what type of computing courses they preferred included teaching styles and size of groups used for projects. The overall result for teaching style preference was project based classes and classes that used examples like live coding demonstrations as the main method for teaching. This was true for all groups that were compared, although computing majors being such a large group forced group projects to be the overall favorite even though non-computing majors choose example based courses by a wide margin. Students also strongly preferred working by themselves or with partner when working on a project. Similar results are shown for all students of different gender, ethnicity, and major. The survey had an option for groups of 6 or more which had 0% of students who preferred working in groups of that size. Avoiding large groups for projects would be beneficial.

Looking back at our hypothesis questions:

Looking at Figure 1, there is an upward trend in overall confidence levels from prior to college to their current college confidence, which helps show that the students are going to continue taking more courses that are computing related.

As shown above, one of the lowest groups of confidence are embodied by females that have taken the survey compared to other groups. Also in that group of low confidence are the Asian students who have taken the survey. Not surprisingly, those who have done a number of computing activities before entering college are much more confident than those who have not.

Answering the questions as to if the student’s peers significantly impact their confidence in computing it can be seen in Figure 15 that the majority of students indicate their confidence is impacted by their peers. Less than 10% of students taking the survey indicated that their confidence was not affected by their peers, clearly there is a correlation. How much of an impact on their confidence there was not able to determined, however it was determined the major cause affecting students’ confidence was their peers knowledge and complexity of coding.

Students’ confidence in their computing ability affected during college classes can be shown in all Figures related to confidence where the delta in confidence between prior and current. In almost all cases there was statistically relevant between prior and current levels of confidence and between current and expected levels of confidence. This shows that student's confidence in their computing ability was affected in a positive way while taking college computing courses.

8. Future Work

While our survey was informative and comprehensive, there is still much to be done in this area of study.

8.1 Followup Survey

In the future, there is room for another survey to expand upon our results. Our study found that students who have taken more computing classes rated their past confidence in their own computer skills higher than students who have taken fewer computing classes. This is an area that can be looked into more thoroughly in future work.

Our survey found that there is a significant increase from prior confidence to current confidence and finally to expected future confidence. This indicates a need for a further study to explore this increase and the reasons behind it.

There are many areas that were not explored. Future work could look into confidence compared to perceived difficulty of classes or how the gender of a professor affects the confidence of his students.

9. Conclusion

In our survey we found trends in our data that fit with our background research about issues at other universities. We found that confidence was something that affect students at WPI, and that factors affecting confidence including how confident students perceived their peers to be. Experience was a big factor in confidence as we would expect it to be, including experience prior to taking any college courses. We recommend tailoring introductory courses to boost confidence in general, and encourage less confident students to engage and participate in class. Teachers can take steps to minimize outspoken students, and encourage answers from students who don't speak as much. Pair programming was found to help out with student retentions, so we recommend continuing to use pair programming in introductory courses, and possibly to use it in even more courses than WPI does right now. In addition to pair programming it is recommended that professors use more example based learning like live coding demonstrations as they were preferred by minority computing students above any other teaching method. For computing assignments and projects these same students also indicated they preferred to work in pairs, but having the option the work alone would also be beneficial. Few students indicated they want to groups of three or larger so minimizing assignments with groups of larger sizes would also be beneficial.

A major focus of the survey was the confidence in computing ability of different groups of people and to determine if students' confidence had increased during the time they had taken college computing courses. For most groups with 95% confidence there was a statistical increase in confidence in computing between prior current and expected. All groups showed the average had increased however some groups had such low number of survey participants that the confidence intervals were massive. This general increase is unsurprising as students confidence in their abilities should increase with the number of computing courses they take. Also unsurprising was confidence was lower for computing minority groups like females, non-computing majors, Asians, and Hispanics/Latinos. These groups lagged behind their male, computing major, and White counterparts through all three time sets (prior, current, expected).

Sources

1. Uolevi Nikula , Orlena Gotel , Jussi Kasurinen, A Motivation Guided Holistic Rehabilitation of the First Programming Course, ACM Transactions on Computing Education (TOCE), v.11 n.4, p.1-38, November 2011
2. Thomas J. Cortina, An introduction to computer science for non-majors using principles of computation, ACM SIGCSE Bulletin, v.39 n.1, March 2007
3. https://www.cs.cmu.edu/afs/cs/project/gendergap/www/papers/sigcse97/sigcse97.html
4. Roli Varma, Making computer science minority-friendly, Communications of the ACM, v.49 n.2, p.129-134, February 2006
5. Don Davis , Timothy Yuen , Matthew Berland, Multiple case study of nerd identity in a CS1 class, Proceedings of the 45th ACM technical symposium on Computer science education, March 05-08, 2014, Atlanta, Georgia, USA
6. Charlie McDowell , Linda Werner , Heather E. Bullock , Julian Fernald, Pair programming improves student retention, confidence, and program quality, Communications of the ACM, v.49 n.8, p.90-95, August 2006
7. IEEE Transactions on Education (Volume: 48, Issue: 2, May 2005)
8. Forte, Andrea, and Mark Guzdial. "Motivation and Non-Majors in Computer Science:." (n.d.): n. pag. *Coweb.cc.gatech.edu*. Georgia Tech. Web. 1 Nov. 2016.
9. http://cra.org/cerp/wp-content/uploads/sites/4/2015/09/Stout-Camp-2015.pdf
10. "Exploring Factors That Influence Computer Science Introductory Course Students to Persist in the Major." *Exploring Factors That Influence Computer Science Introductory Course Students to Persist in the Major* (n.d.): n. pag.*Barker-McDowell-Kalahar*.
11. Lori Carter, Why students with an apparent aptitude for computer science don't choose to major in computer science, Proceedings of the 37th SIGCSE technical symposium on Computer science education, March 03-05, 2006, Houston, Texas, USA
12. Ani Nahapetian, La bella figura: making a good impression when teaching an introduction to programming to non-engineers, Journal of Computing Sciences in Colleges, v.26 n.4, p.122-129, April 2011
13. Robert Bryant , Judith Cushing , Jenny Orr , Matthew Dickerson , Richard Weiss, The impact of computing courses for non-majors, Journal of Computing Sciences in Colleges, v.30 n.1, p.176-177, October 2014
14. Lilly Irani, Understanding gender and confidence in CS course culture, ACM SIGCSE Bulletin, v.36 n.1, March 2004
15. Lecia J. Barker , Melissa O'Neill , Nida Kazim, Framing classroom climate for student learning and retention in computer science, Proceedings of the 45th ACM technical symposium on Computer science education, March 05-08, 2014, Atlanta, Georgia, USA